

# Knowledge Completion for Generics using Guided Tensor Factorization

Hanie Sedghi and Ashish Sabharwal

Allen Institute for Artificial Intelligence (AI2)  
Seattle, WA, 98103, U.S.A.

hanies,ashishs@allenai.org

February 27, 2017

## Abstract

Given a knowledge base (KB) rich in facts about common nouns or generics, such as “all trees produce oxygen” or “some animals live in forests”, we consider the problem of deriving additional such facts at a high precision. While this problem has received much attention for named entity KBs such as Freebase, little emphasis has been placed on generics despite their importance for capturing general knowledge. Different from named entity KBs, generics KBs involve implicit or explicit quantification, have more complex underlying regularities, are substantially more incomplete, and violate the commonly used locally closed world assumption (LCWA). Consequently, existing completion methods struggle with this new task. We observe that external information, such as relation schemas and entity taxonomies, if used correctly, can be surprisingly powerful in addressing the challenges associated with generics. Using this insight, we propose a simple yet effective knowledge guided tensor factorization approach that achieves state-of-the-art results on two generics KBs for science, doubling their size at 74%-86% precision. Further, to address the paucity of facts about rare entities such as oriole (a bird), we present a novel taxonomy guided submodular active learning method to collect additional annotations that are over five times more effective in inferring further new facts than multiple active learning baselines.

## 1 Introduction

We consider the problem of completing a partial knowledge base (KB) containing facts about generics or common nouns, represented as a third-order tensor of  $(source, relation, target)$  triples, such as  $(butterfly, pollinate, flower)$ . Since generics represent classes of similar individuals, the truth value  $y_i$  of a generics triple  $x_i = (s, r, t)$  depends on the quantification semantics one associates with  $s$  and  $t$ . A natural setting, especially relevant for noisy facts derived automatically from text, is to associate  $s$  with a categorical quantification from  $\{all, some, none\}$  and associate  $t$  (implicitly) with *some*. For instance, “all butterflies pollinate (some) flower” and “some animals live in (some) forest”. We treat the quantification of  $s$  as the categorical label  $y_i$  for the triple  $x_i$ . Given a KB of such triples, the task is to infer as many more triples as possible at a high (e.g., 80%) precision.

Tensor factorization and graph based methods have both been found to be very effective for expanding knowledge bases, but have focused on *named entity* KBs such as Freebase [2] involving relations with clear semantics such as *bornIn* and *isACityIn* and disambiguated entities such as *Barack Obama* or *Hawaii*. It has been observed that there are many reliable Horn clauses that

connect predicates in this setting. For instance, for any person  $x$ , city  $y$ , and country  $z$ ,  $(x, \text{bornIn}, y) \ \& \ (y, \text{isACityIn}, z) \Rightarrow (x, \text{nationalityIs}, z)$ . With generics, however, clear patterns or reliable first-order logic rules are rare, in part due to each generic representing a collection of entities that often have similarities and differences w.r.t. various relations. For example,  $(x, \text{liveIn}, \text{mountain})$  is true for many *cats* and *caribou*, but there is little tangible similarity between the two animals.

On the other hand, generics often come with additional rich background knowledge complementing the information present in the KB itself, such as a *taxonomic hierarchy* of entities (available from sources such as WordNet [15]) and the corresponding *entity types* and *relation schema*.

Our key insight is that, if used appropriately, *taxonomic and schema information can be surprisingly effective in making tensor factorization methods vastly more effective for generics* for deriving high precision facts. Intuitively, many properties of generics tend to be shared by their siblings in the taxonomy (e.g., finch, oriole, and hummingbird) where as siblings of named entities (e.g., various people) differ widely (e.g., who they are married to, where they live, etc.). We propose three ways of using this information and empirically demonstrate the effectiveness of each for generics.

First, we observe that simply imposing schema consistency (Section 3.1) on derived facts can boost the performance of state-of-the-art methods such as Holographic Embeddings (HolE) [19] from nearly no new facts at 80% precision to over 10,000 new facts, starting with a generics KB of about the same size. Other low-dimensional embedding methods, such as TransE [3], RESCAL [17], and CNTF [20] (which uses schema information as well), also obtained no new facts at 80% precision. Graph-based completion methods did not scale to our densely connected tensors.<sup>1</sup>

Second, one can further boost performance by transferring knowledge up and down the taxonomic hierarchy, using the quantification semantics of generics (Section 3.2). We show that simply expanding the starting tensor this way before applying tensor factorization results in a statistically significant higher precision over new facts at the same yield.

Finally, we propose a novel limited-budget taxonomy guided active learning method to address the challenge of significant incompleteness in generics KBs, by quantifying uncertainty via siblings (Section 4). Reliable facts about generics are much harder to derive using state-of-the-art information extraction methods than facts about named entities. The large number of many-to-many relations makes it challenging to cover most of the true facts by only using textual information from books, the Web, or other corpora. This, in turn, makes generics KBs vastly incomplete, with no or very little information about certain entities such as caribou or oriole.

To predict facts about a previously unseen or rarely seen entity  $\tilde{e}$ , we first identify and annotate a small number of active queries about  $\tilde{e}$  and then feed this to the factorization framework to infer more facts about  $\tilde{e}$ . Specifically, we define a correlation based measure of the uncertainty of each triple involving  $\tilde{e}$ , based on how frequently is the corresponding triple true for  $\tilde{e}$ 's siblings in the taxonomic hierarchy (Section 4.1). We then propose a submodular objective function, and a corresponding greedy  $(1 - 1/e)$ -approximation, to search for a small subset of triples to annotate that optimally balances diversity with coverage (Section 4.2). We demonstrate that annotating this balanced subset makes tensor factorization derive substantially more new and interesting facts compared to several active learning baselines.

## 1.1 Related Work

KB completion approaches fall into two main classes: graph-based methods and those employing low-dimensional embeddings via matrix or tensor factorization [8, 18, 21]. The former use graph

<sup>1</sup> On the smaller Animals tensor (to be described later), PRA [13] generated very few high-precision facts after 30 hours. SFE [9], which typically produces better results than PRA, was unable to finish training a classifier for any relation after a day. On the other hand, HolE is trained in a couple of minutes even on the larger Science tensor.

traversal techniques to complete the KB [9, 13]. While highly effective on named entity KBs, this class of solutions doesn’t scale well to our setting (cf. Footnote 1). This may be due to different connectivity characteristics of generics tensors compared to named entity ones such as FB15k [3].

Several embedding-based methods have been highly successful at KB completion. We compare against many of these, including variants of HolE, TransE, and RESCAL.

Recent work on incorporating entity type and relation schema in tensor factorization [11, 12, 25] has focused on factual databases, with very different characteristics than generics tensors. Nimishakavi et al. [20] use entity type information as a matrix in the context of non-negative RESCAL for schema induction on medical research documents. As a byproduct, they complete missing entries in the tensor in a schema-compatible manner. We show that our proposal performs better on generics tensors than their method, CNTF. Chang et al. [4]’s TRESICAL system also incorporates types in RESCAL in a manner very similar to CNTF.

For schema-aware discriminative training of embeddings, Xie et al. [25] use a flexible ratio of negative samples from both schema consistent and schema inconsistent triples. Their combined ideas, however, do not improve upon vanilla HolE (one of our baselines) on the standard FB15k [3] dataset. They also consider imposing hierarchical types for Freebase, as entities may have different meanings when they have different types – an issue that typically does not apply to generics KBs. Incorporating given first order logic rules (which are unavailable for generics) has been explored for the simpler case of matrix factorization [7, 22].

Xie et al. [24] consider inferring facts about a new entity  $\tilde{e}$  given a ‘description’ of that entity. They use Convolutional Neural Networks (CNNs) to encode the description, deriving an embedding for  $\tilde{e}$ . Such a description in our context would correspond to knowing some factual triples about  $\tilde{e}$ , which is a restricted version of our active learning setting.

Krishnamurthy and Singh [10] consider active learning for CP decomposition of tensors. They start with an *empty* tensor and look for the most informative slices and columns to fill *completely* to achieve optimal sample complexity. The *incoherence* assumption on the column space that their framework builds upon, however, does not apply to generics KB.

## 2 Tensors of Generics

We consider knowledge expressed in terms of *(source, relation, target)* triples, abbreviated as  $(s, r, t)$ . Such a triple may refer to *(subject, predicate, object)* style facts commonly used in information extraction. Each source and target is an *entity* that is a generic noun, e.g., animals, habitats, or food items. Examples of relations include *foundIn*, *eat*, etc. With each generics triple  $(s, r, t)$ , we associate a categorical truth value  $q \in \{all, some, none\}$ , defining the quantification semantics “ $q$   $s$   $r$  (some)  $t$ ”. For instance, “some animals live in (some) forest” and “all dogs eat (some) bone”. *Given a set  $K$  of such triples with annotated truth values, the task is to predict additional triples  $K'$  that are also likely to be true.*

In addition to a list of triples, we assume access to background information in the form of entity types and the corresponding *relation schema*, as well as a *taxonomic hierarchy*.<sup>2</sup> Let  $E_T$  denote the set of possible entity types. For each relation  $r$ , the relation schema imposes a type constraint on the entities that may appear as its source or target. Specifically, the *schema* for  $r$  is a collection  $\mathcal{S}_r = \{(\mathcal{D}_r^{(i)}, \mathcal{R}_r^{(i)}) \subseteq E_T \times E_T \mid i \in [\ell]\}$  of domain-range pairs such that the truth value of  $(s, r, t)$  is *none* whenever for every  $i \in [\ell]$  it is the case that  $s \notin \mathcal{D}_r^{(i)}$  or  $t \notin \mathcal{R}_r^{(i)}$ . For example, the relation *foundIn* may be associated with the schema  $\mathcal{S}_{foundIn} = \{(animal, location), (insect, animal), (plant, habitat), \dots\}$ . Similarly, the taxonomic hierarchy defines a partial order  $\mathcal{H}$  over all entities that

<sup>2</sup>We do not assume the schema or taxonomy is perfect, and instead rely on these only for heuristic guidance.

captures the “isa” relation, with direct links such as *isa(dog, mammal)* or *isa(gerbil, rodent)*. We use this information to extract “siblings” of a given entity, i.e., entities that share a common parent (this may be easily generalized to any common ancestor).

### 3 Guided Knowledge Completion

We begin with an overview of tensor factorization for KB completion for generics. Let  $(s, r, t)$  be a generics triple associated with a categorical quantification label  $q \in \{all, some, none\}$ . For example,  $((cat, havePart, whiskers), all)$ ,  $((cat, liveIn, homes), some)$ , and  $((cat, eat, bear), none)$ . Predicting such labels is thus a multi-class classification problem. Given a set  $K$  of labeled triples, the goal of tensor factorization is to learn a low-dimensional embedding  $h$  for each entity and relation such that some function  $f$  of  $h$  best captures the given labels. Given a new triple, we can then use  $f$  and the learned  $h$  to predict the probability of each label for it.  $K$  often contains only “positive” triples, i.e., those with label *all* or *some*. A common step in discriminative training for  $h$  is thus *negative sampling*, i.e., generating additional triples that (are expected to) have label *none*.

Let  $K = \{(x_i, y_i), i \in [m]\}$  be a set of triples  $x_i = (s_i, r_i, t_i)$  and corresponding labels  $y_i \in \{1, 2, 3\}$  equivalent to categorical quantification label  $q_i \in \{all, some, none\}$ . We learn entity and relation embeddings  $\Theta$  that minimizes the multinomial logistic loss as:

$$\begin{aligned} & \min_{\Theta} \sum_{i=1}^m \sum_{k=1}^3 -\mathbb{1}\{y_i = k\} \log \Pr(y_i = k \mid x_i, \Theta) \\ & = \min_{\Theta} \sum_{i=1}^m \sum_{k=1}^3 -\mathbb{1}\{y_i = k\} \log \sigma(y_i f(h_r, h_s, h_t)), \end{aligned} \quad (1)$$

where  $h_r, h_s, h_t \in \mathbb{R}^d$  denote the learned embeddings (latent vectors) for  $s, r, t$ , respectively, and  $\sigma(\cdot)$  is the sigmoid function defined as  $\sigma(z) = \frac{1}{1+\exp(-z)}$ .

If the *all* categorical label for generics is unavailable, we can simplify the label space to  $\{some, none\}$ , modeled as  $y_i \in \{\pm 1\}$ , and reduce the model to binary classification:

$$\min_{\Theta} \sum_{i=1}^m \log [1 + \exp [-y_i f(h_r, h_s, h_t)]] . \quad (2)$$

While all our proposed schemes are embedding oblivious, for concreteness, we describe and evaluate them for the recent Holographic Embeddings (HolE) model [19] which models the label probability as:

$$f(h_r, h_s, h_t) = h_r^\top (h_s \circ h_t) \quad (3)$$

where  $\circ : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^d$  denotes circular correlation:

$$[a \circ b]_k = \sum_{i=0}^{d-1} a_i b_{(i+k) \bmod d} . \quad (4)$$

As can be deduced from eqns. (3)-(4), this model can capture asymmetry of relations. In addition, circular correlation can be performed using the fast Fourier transform (FFT), making HolE highly efficient.

### 3.1 Incorporating Types, Relation Schema (ITRS)

As described earlier, relation schema  $\mathcal{S}_r$  imposes a restriction on sources and targets that may occur with a relation  $r$ . We can incorporate this knowledge both at training and at test times. Doing this at test time simply translates to relabeling schema inconsistent predicted triples as *none*. Incorporating this knowledge at training time can be done as a constraint on the random negative samples that the method generates to complement the given, typically positive, triples for training. In general, the ratio of random negative samples from the entire tensor  $\mathcal{T}$  and random negative samples from the schema consistent portion  $\mathcal{T}'$  of  $\mathcal{T}$  is a parameter that should be tuned such that the resulting negative samples mimic the true distribution of labels. It is worth noting that the locally closed world assumption (LCWA) plays an important role in determining this ratio. However, this idea has been used in the literature without considering the nature of the dataset, resulting in some seemingly contradicting empirical results on the optimal ratio [14, 25]. As discussed later, we found sampling from  $\mathcal{T}$  worked best on our datasets.

### 3.2 Incorporating Entity Taxonomy (IET)

It is challenging to come up with complex Horn or first order logic rules for generics, as each entity represents a class of individuals that may not all behave identically. However, we can derive simple yet highly effective rules based on categorical quantification labels, leveraging the fact that entities come from different levels in a taxonomy hierarchy. Let  $p$  be the parent entity for entity set  $\{c_i\}$ . Then we have:

$$\begin{aligned} ((p, r_j, t_j), all) &\Rightarrow \forall i \ ((c_i, r_j, t_j), all) \\ \forall i \ ((c_i, r_j, t_j), all) &\Rightarrow ((p, e_j, t_j), all) \\ \exists i \ ((c_i, r_j, t_j), all) &\Rightarrow ((p, e_j, t_j), some) \\ \exists i \ ((c_i, r_j, t_j), some) &\Rightarrow ((p, e_j, t_j), some) \end{aligned}$$

We apply these rules to address sparsity of generics tensors, making tensor factorization more robust. Specifically, given initial triples  $K$ , we use applicable rules to derive additional triples  $K'$ , perform tensor factorization on  $K \cup K'$ , and then revisit the triples in  $K'$  using their predicted label probabilities. Note that this approach allows us to be robust to taxonomic errors: instead of assuming each triple in  $K'$  is true, we use this only as a prior and let tensor factorization determine the final prediction based on global patterns it finds.

## 4 Active Learning for New or Rare Entities

To address the incomplete nature of generics KBs, we consider *rare* entities for which we have very few facts, or *new* entities which are present in the taxonomy but for which we have no facts in the KB. The goal is to use tensor factorization to generate high quality facts about such entities. For instance, consider the task of inferring facts about *oriole*, where *all we know is that it is a bird*. We assume a restricted budget on number of the facts we can query (for human annotation) about *oriole*, using which we would like to predict many more high-quality facts about it.

Given a fixed query budget  $B$ , what is the optimal set queries we should generate for human annotation about a new or rare entity  $\tilde{e}$  for this task? We view this as an *active learning* problem and propose a two-step algorithm. First, we use taxonomy guided uncertainty sampling to propose a list  $L$  to potentially query. Next, we describe a submodular objective function and a corresponding linear time algorithm to choose an optimal subset  $\hat{L} \subseteq L$  satisfying  $|\hat{L}| = B$ . We then use  $\hat{L}$  for

human annotation, append the result to the original KB, and perform tensor factorization to predict additional new facts about  $\tilde{e}$ . For notational simplicity and without loss of generality, throughout this section, we consider the case where  $\tilde{e}$  appears as the source entity in the triple; the ideas apply equally when  $\tilde{e}$  appears as the target entity in the triple.

#### 4.1 Knowledge Guided Uncertainty Quantification

We now discuss the active learning and specifically uncertainty sampling method we use to propose a list of triples to query. Uncertainty sampling considers the uncertainty for each possible triple  $(\tilde{e}, r_i, e_i)$ , defined as one minus the conditional probability of this fact given the facts we already know (i.e., KB) [23]. The question is how to model this conditional probability. A simple baseline is to consider **Random** queries, i.e.,  $r, e$  are selected randomly from the list of relations and entities in the tensor, respectively.

To infer information about  $\tilde{e}$ , we propose the following approximation for the conditional probability of a new fact about  $\tilde{e}$  given the KB. Let  $\tilde{E}_{\tilde{e}} = \{e \mid \text{corr}(\tilde{e}, e) > 0\}$  be set of entities that are correlated with  $\tilde{e}$  and  $\Omega = \{(e_i, r_i, e'_i), y_i \mid e_i \in \tilde{E}_{\tilde{e}}\}$  be set of known facts about such entities and  $y_i$  be the label for triple  $(e_i, r_i, e'_i)$ . We have:

$$\Pr(\phi_{r_i}(\tilde{e}, e'_i), 1) \simeq \frac{1}{|\Omega|} \sum_{e_i \in \tilde{E}_{\tilde{e}}} \text{corr}(\tilde{e}, e_i) y_i \quad (5)$$

However, in practice, we cannot measure  $\text{corr}(\tilde{e}, e_i)$  for every entry in the KB as we do not have complete information about  $\tilde{e}$ . One simple idea is to consider that every entity is correlated with  $\tilde{e}$ :  $\text{corr}(\tilde{e}, e_i) = 1 \ \forall e_i \in E$ . We will refer to this as **Schema Consistent** query proposal as this relates to summing over all possible (hence schema consistent) facts. Since we have access to taxonomy information, we can do a more precise, **Sibling Guided**, approximation. We propose the following approximation for  $\text{corr}(\tilde{e}, e_i), e_i \in E$ ,

$$\text{corr}(\tilde{e}, e_i) = \begin{cases} 1 & e_i \in \text{sibling}(\tilde{e}) \\ 0 & \text{o.w.} \end{cases} \quad (6)$$

Eqns. (5) and (6) can be used to infer uncertain triples: if every sibling of  $\tilde{e}$  has relationship  $r$  with an entity  $e'$ , we can infer this is the case for  $\tilde{e}$  as well. On the other hand, when siblings disagree in this respect, there is more uncertainty about  $(\tilde{e}, r, e')$  (according to (5) and (6)), making this triple a good candidate to query. Setting some upper and lower bound on the uncertainty (conditional probability as in (5)), we reach a set  $L = \{(\tilde{e}, r_i, e_i), i \in I\}$  of triples to query. We also infer the set  $M = \{(\tilde{e}, r_j, e_j), j \in J\}$  of triples that a large majority of siblings agree upon, and hence  $\tilde{e}$  is expected to agree as well. See Algorithm 1. In our example of *oriole*, the siblings are the *birds* that exist in the tensor, e.g., *hummingbird*, *finch*, *hummingbird*, *woodpecker*, etc. All of them (*eat*, *insect*) and hence we infer this for oriole. But there is no agreement on (*appearIn*, *farm*) and hence this is added to the query list.

#### 4.2 Efficient Subset Selection

Given the list  $L$  as above (Algorithm 1), which we can write in short as  $L = \{(r_i, e_i), i \in I\}$ , the problem is to find the “best” subset  $\hat{L}$ . A baseline for such a selection is to choose the top  $k$  queries. We will refer to this as **TK** subset selection.

Viewing subset selection as a combinatorial problem, we devise an objective  $\mathcal{F}$  that models several natural properties of this subset. We then prove that  $\mathcal{F}$  is **submodular**, implying that a simple



---

**Algorithm 1:** Active Learning for Query Proposal

---

**input** new entity  $\tilde{e}$ , KB, taxonomy, lower bound  $\kappa_M$  on agreement, lower bound  $\tau_L$  on uncertainty, upper bound  $\tau_U$  on uncertainty  
1: Extract list  $S_{\tilde{e}}$  of sibling( $\tilde{e}$ ) using taxonomy  
2: For each  $e_i \in S_{\tilde{e}}$ , add all facts about  $e_i$  to  $\Omega$   
3: **for**  $(\tilde{e}, r_i, e'_i) \in \Omega$  **do**  
4:   Use (5) and (6) to estimate  $p = \Pr(\phi_{r_i}(\tilde{e}, e'_i) = 1)$   
5:   **if**  $p \geq \kappa_M$  **then** Add  $(\tilde{e}, r_i, e'_i)$  to M  
6:   **if**  $\tau_L \leq p \leq \tau_U$  **then** Add  $(\tilde{e}, r_i, e'_i)$  to L  
**output** L, M

---

greedy algorithm can efficiently compute a  $(1 - 1/e)$ -approximation of the global optimum [16]. We refer to this as **SM** subset selection method.

Since queried samples will eventually be fed into tensor factorization, we would like  $\hat{L}$  to *cover* entities (for the other argument of the triple) and relations as much as possible. In addition, we would like  $\hat{L}$  to be *diverse*, i.e., prioritize relations and entities that are more varied.<sup>3</sup> At the same time, we would also want to minimize redundancy, i.e., avoid choosing relations (entities) that are too similar. Let  $\mathcal{F}(\hat{L}, R_{\hat{L}}, E_{\hat{L}})$  denote our objective, where  $R_{\hat{L}}, E_{\hat{L}}$  is the set of relations and entities in  $\hat{L}$ , respectively. We decompose it as:

$$\mathcal{F}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) = w_{\mathcal{C}}\mathcal{C}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) + w_{\mathcal{D}}\mathcal{D}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) - w_{\mathcal{R}}\mathcal{R}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) \quad (7)$$

where the terms in RHS correspond to *coverage*, *diversity* and *redundancy*, resp., and  $w_{\mathcal{C}}, w_{\mathcal{D}}, w_{\mathcal{R}}$  are the corresponding non-negative weights. Next, we propose functional forms for these terms. Note that any function that captures the described properties can be used instead, as long as the objective remains submodular.

Let  $R$  and  $E$  denote the set of relations and entities in the KB, resp. The coverage simply captures the fraction of entity and relations that we have included in  $\hat{L}$ :

$$\mathcal{C}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) = \frac{|R_{\hat{L}}|}{|R|} + \frac{|E_{\hat{L}}|}{|E|}.$$

The diversity for  $\hat{L}$  is the sum of the diversity measure of the entities and relations included in the set:

$$\mathcal{D}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) = \sum_{(r,e) \in \hat{L}} [V_r + V_e].$$

We define the diversity measure for each relation  $r$  as the ratio of the number of entities that appear in the KB as its source or target over the number all entities. For an entity, we define the notion of diversity in the same manner as shown below. Note that diversity measure is an intrinsic characteristic of each entity and relationship dictated by KB and is independent of the set  $L$ . Let  $V_r$  and  $V_e$  represent the diversity measure of relation  $r$  and entity  $e$ , respectively. We use  $E_{S_r}, E_{T_r}$  to denote the set of sources and targets that appear for relation  $r$  in the KB and  $R_e$  as the set of

---

<sup>3</sup>This agrees with the sampling method of Chen et al. [5] for factorizing coherent *matrices* with missing values, which chooses samples with probability proportional to their local coherence.

---

**Algorithm 2:** Subset Selection for Query Selection

---

**input** KB, B as budget size, L from Algorithm 1.  
1:  $\forall (r, e) \in L$ , compute the diversity measure  $V_r, V_e$   
2:  $\hat{L} \leftarrow \emptyset$   
3: **for**  $j = 1$  to  $B$  **do**  
4:  $\forall l \in L \setminus \hat{L} : \mathcal{G}(l) = \mathcal{F}(\hat{L} \cup l) - \mathcal{F}(\hat{L})$ , for  $\mathcal{F}$  in (7)  
5: Select  $l^* = \arg \max_{L \setminus \hat{L}} \mathcal{G}(l)$   
6: Add  $l^*$  to  $\hat{L}$   
**output**  $\hat{L}$

---

relations in the KB that have  $e$  as their target and  $E_{S_e}$  as the set of entities that appear as the first entity when  $e$  is the second entity of the triple in the KB,

$$V_r = \frac{|E_{S_r}| + |E_{T_r}|}{|E|}, \quad V_e = \frac{|R_e| + |E_{S_e}|}{|R| + |E|}.$$

As described above, redundancy is a measure of similarity between relations(entities) in  $\hat{L}$ . Tensor factorization yields an embedding for each relation(entity) given the facts they participated in. Therefore, the learned embeddings are one of the best options for capturing similarities. Let  $h_e(h_r)$  denote the learned embedding for entity  $e$ (relation  $r$ ). We define

$$\mathcal{R}(\hat{L}, R_{\hat{L}}, E_{\hat{L}}) = \sum_{r_1, r_2 \in \hat{L}} \|h_{r_1} - h_{r_2}\| + \sum_{e_1, e_2 \in \hat{L}} \|h_{e_1} - h_{e_2}\|.$$

**Theorem 1.** *Algorithm 2 gives a  $(1 - 1/e)$ -approximation of the global optimum of  $\mathcal{F}$ .*

Theorem 1 follows by proving that the objective  $\mathcal{F}$  in eqn. (7) is submodular. Since addition preserves submodularity and the weights  $w_{\mathcal{L}}, w_{\mathcal{D}}, w_{\mathcal{R}}$  are non-negative, it suffices to show that each term in  $\mathcal{F}$  is submodular. For proof, see Appendix A. We compare our query proposal and subset selection methods with the corresponding baselines in Table 2.

## 5 Experiments

**Dataset and Setup:** To assess the quality of our guided KB completion method, we consider two knowledge bases about generics: (1) a **Science** tensor containing facts about animals, occupations, activities, locations, etc. [6]; and (2) its **Animals** sub-tensor that focuses on facts about animals. This data does not include  $((s, r, t), all)$  style triples, so we use the objective function in Eqn. (2) rather than the multi-class one in Eqn. (1). Dalvi et al. use a pipeline consisting of Open IE [1] extractions, aggregation, and clean up via crowd-sourcing to generate domain-targeted facts about elementary level science. These facts come with a relevant WordNet [15] based taxonomy, entity types, and relation schema. Table 1 summarizes statistics of the datasets we use for experimentation,<sup>4</sup> where we ensure every entity is mentioned at least 20 times in the corresponding tensor.<sup>5</sup> In order to limit error propagation, we use a two-level abstraction of the taxonomy.

---

<sup>4</sup>Datasets and code available from the authors.

<sup>5</sup>This frequency filtering is a milder version of the mainstream approach for KB completion. E.g., FB15K uses a threshold of 100.



Dataset	# Entities	# Relations	# Triples
Animals	224	129	10,604
Science	1,255	1,513	66,643

Table 1: Datasets. We use a 3/1/1 train/validation/test split.

**Guided KB Completion:** We first compare our method (Section 3) with existing KB completion techniques on the Animals tensor, and then demonstrate that its effectiveness carries over scalably to the larger Science tensor as well.

We examine two alternatives for generating negative samples: given a triple  $(s, r, t) \in \mathcal{T}$ , replace  $s$  with (1) any entity  $s'$  or (2) an entity  $s'$  of the same type as  $s$ . The resulting perturbed triple  $(s', r, t)$  is then treated as a negative sample if it is not present in  $\mathcal{T}$ . We also considered a weighted combination of (1) and (2), and found random sampling to be the most reliable on our datasets. This complies with lack of LCWA.

We consider extensions of three state-of-the-art embedding-based KB completion methods: HolE, TransE, and RESCAL. As mentioned earlier, two leading graph-based methods, SFE and PRA, did not scale well. Both vanilla TransE and RESCAL resulted in poor performance; we report numbers only for their extensions. Specifically, we consider 3 baselines: (1) HolE, (2) TransE+Schema, and (3) CNTF which extends RESCAL and incorporates schema.

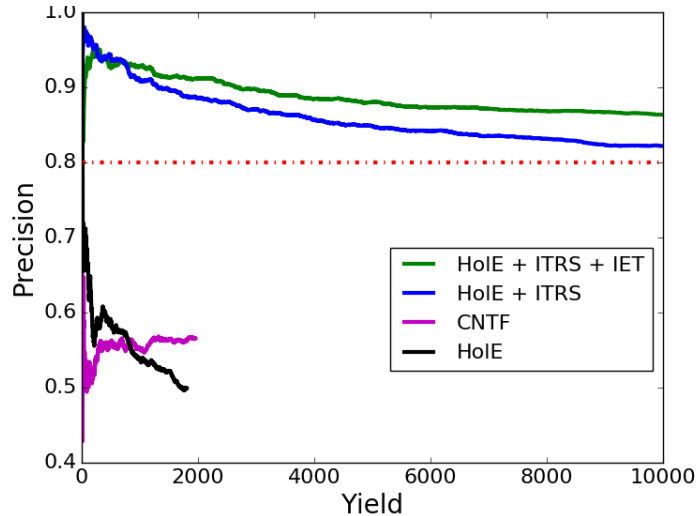


Figure 1: Precision-yield curves for various embedding-based methods on the Animals tensor.

Figure 1 shows the resulting precision-yield curves for the predictions made by each method on the **Animals** dataset containing 10.6K facts. TransE+ITRS gave a precision of only around 10% and is omitted from the plot. We make two observations. First, deriving new facts for these generics tensors at a high precision is challenging! Specifically, none of the baseline methods achieve a yield of more than 10% of  $\mathcal{T}$  with precision at least 60%. Second, external information, if used appropriately, can be surprisingly powerful in this setting. Specifically, simply incorporating relation schema (ITRS) allows HolE-based completion to double the size of the starting tensor  $\mathcal{T}$  by producing over 10K new triples at a precision of 82%. Further, incorporating entity taxonomy (IET) to address tensor sparsity results in the same yield at a statistically significantly higher precision of 86.4%.

Figure 2 lists the top 20 predictions by various approaches, illustrating that our proposed

HoIE	CNTF	HoIE+IET+ITRS
<p>farm,join,farm  family,join,family  tree,resemble,tree  water,is known as,water  virus,attract,virus  animal,resemble,animal  tree,is known as,tree  habitat,is known as,habitat  environment,is known as,environment  man,join,man  bird,give birth to,bird  region,is known as,region  virus,derive from,virus  food,resemble,food  bird,is known as,bird  field,resemble,field  fish,is known as,fish  bird,resemble,bird  grass,graze in,man  animal is known as animal</p>	<p>penguin,has part, tooth  mosquito,spread,parasite  spider,has part,skin  elephant,eat,fish  shark,has part,skin  crab,eat,insect  snake,eat,fish  otter,has part,tooth  meat,attract,hummingbird  spider,has part,claw  turtle,has part,tooth  human,eat,plant  monkey,has part,wing  dolphin,has part,tooth  carnivore,live in,water  lizard,eat,fish  pelican,has part,tooth  caterpillar,turn into,bird  bee,pollinate,garden  virus,infect,bird</p>	<p>salmon,thrive in,water  animal,give birth to,animal,  duck,feed in,water  fish,migrate to,water  fish,thrive in,water  turtle,swim in,water  salmon,swim in,water  turtle,live in,water  animal,chew,food  insect,destroy,tree  farm,possess,horse  fish,swim in,ocean  turtle,feed in,water  turtle,float in,water  dinosaur,walk on,leg  turtle,migrate to,water  turtle,return to,water  man,ride,cattle  turtle,swim in,ocean,  fish,float in,ocean</p>

Figure 2: Top 20 predictions by various methods. True triples are shown in black and false triples in red.

method infers many interesting and useful generic facts about the world.

Next, we evaluate our proposal on the entire **Science** dataset with 66.6K facts. Since graph-based methods did not scale well to the much smaller Animals dataset and other methods performed substantially worse there, we focus here on the scalability and prediction quality of our method. We found that HoIE+ITRS+IET scales well to this high dimension, doubling the number of facts by adding 66K new facts at 74% precision. Although the Science tensor is 1,000 times larger than the Animals tensor, the method took only 10x longer to run (3 minutes on Animals tensor vs. 56 minutes on Science tensor, using a 2.8GHz, 16GB Macbook Pro).

**Active Learning for New Entities:** To assess the quality of our active learning mechanism (Section 4), we consider predicting facts about a new entity  $\tilde{e}$  that is not in the Animals tensor. For illustration, we choose  $\tilde{e}$  from the Science vocabulary while ensuring that it is present in the WordNet taxonomy. The setup is as follows. We first use a *query generation mechanism* (Random, Schema Consistent, or Sibling Guided; cf. Section 4.1) to propose an ordered list  $L$  of facts about  $\tilde{e}$  to annotate. Next, we perform *subset selection* (Top K or TK, Submodular or SM; cf. Section 4.2) on  $L$  to identify a subset  $\hat{L}$  of up to 100 most promising queries. These are then annotated and the true ones fed into tensor factorization as additional input to infer further new facts about  $\tilde{e}$ .

In Table 2, we assess the quality of  $\hat{L}$  in two ways, when  $|\hat{L}| = 100$ : how many true facts does  $\hat{L}$  have and how many overall new facts does this annotation produce about  $\tilde{e}$ . Figure 3 provides a complementary view, focusing on the overall number of new facts inferred as  $|\hat{L}|$  increases. While these illustrative numbers are for a representative new entity, *reindeer*, the overall trend and order of numbers remain the same.

We highlight a few observations. First, not surprisingly, randomly choosing triples about  $\tilde{e}$  to annotate is ineffective. Second, choosing schema consistent triples results in 73 true triples (out of 100) but these facts help tensor factorization very little, resulting in only 10 additional new triples about  $\tilde{e}$ . Our proposed sibling guided querying mechanism results not only in nearly all 100 facts being true along with 17 true facts inferred from sibling agreement (set M in Algorithm 1), but also,

Query Proposal	Subset Selection	# New True Triples Inferred			
		From Annt'n	Sibling Agrm't	Tensor Fact'n	Total
Random	-	0	-	0	0
Schema Consistent	TK	73	-	10	83
Schema Consistent	SM	57	-	27	84
Sibling Guided	TK	96	17	211	324
<b>Sibling Guided</b>	<b>SM</b>	100	17	366	<b>483</b>

Table 2: Active Learning for new entities: Number of new facts inferred (from annotation, sibling agreement, tensor factorization, and in total) for a representative new entity  $\tilde{e}$ , when querying 100 facts about  $\tilde{e}$  for human annotation.

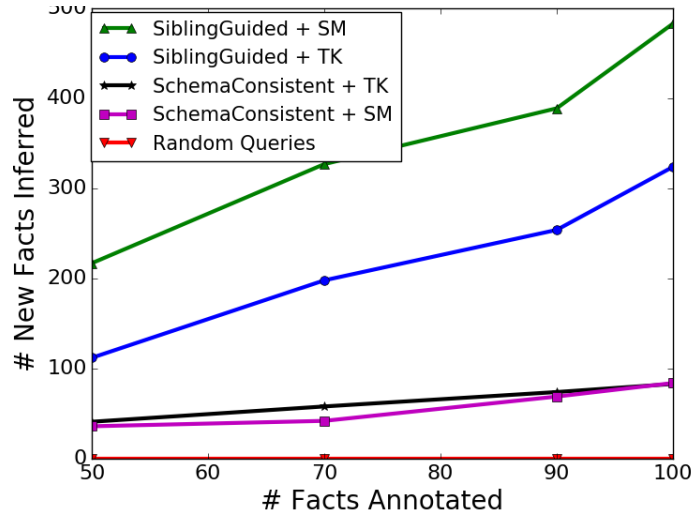


Figure 3: Active Learning for new entities: Total number of new inferred facts for various human annotation query sizes.

combined with submodular subset selection for balancing diversity with coverage (Algorithm 2), ultimately results in 483 new facts about  $\tilde{e}$ , including reindeer eat fruit, wolf chase reindeer, and reindeer provide fur. The plot in Figure 3 demonstrates that the qualitative trends remain the same, irrespective of the number  $|\hat{L}|$  of queries annotated. Overall, our sibling guided queries with submodular subset selection ultimately results in 5.8 times more new facts about  $\tilde{e}$  than a non-trivial, uncertainly based, schema consistent baseline.

## 6 Conclusion

This work explores KB completion for a new class of problems, namely completing generics KBs, which is an essential step for including general world knowledge in intelligent machines. The differences between generics and much studied named entity KBs make existing techniques either not scale well or produce facts at an undesirably low precision out of the box. We demonstrate that incorporating entity taxonomy and relation schema appropriately can be highly effective for generics KBs. Further, to address scarcity of facts about certain entities in such KBs, we devise a novel active learning approach using sibling guided uncertainty estimation along with submodular subset selection. The proposed techniques substantially outperform various baselines, setting a new

state of the art for this challenging class of completion problems.

## Acknowledgments

The authors would like to thank Peter Clark for fruitful discussions, valuable feedback, and crowd-sourcing annotations; Matt Gardner for constructive comments and assessing graph-based completion methods on our datasets; and Udai Saini and Partha Talukdar for evaluating their CNTF approach on our datasets.

## References

- [1] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open information extraction from the web. In *IJCAI*, 2007.
- [2] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.
- [3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795, 2013.
- [4] Kai-Wei Chang, Wen tau Yih, Bishan Yang, and Christopher Meek. Typed tensor decomposition of knowledge bases for relation extraction. In *EMNLP*, 2014.
- [5] Yudong Chen, Srinadh Bhojanapalli, Sujay Sanghavi, and Rachel Ward. Coherent matrix completion. *ICML*, 2014.
- [6] Bhavana Dalvi, Niket Tandon, and Peter Clark. Domain-targeted, high precision knowledge extraction. *TACL*, 2017. To appear.
- [7] Thomas Demeester, Tim Rocktäschel, and Sebastian Riedel. Lifted rule injection for relation embeddings. In *EMNLP*, 2016.
- [8] Xin Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM, 2014.
- [9] Matthew Gardner and Tom M. Mitchell. Efficient and expressive knowledge base completion using subgraph feature extraction. In *EMNLP*, 2015.
- [10] Akshay Krishnamurthy and Aarti Singh. Low-rank matrix and tensor completion via adaptive sampling. In *Advances in Neural Information Processing Systems*, pages 836–844, 2013.
- [11] Denis Krompaß, Maximilian Nickel, and Volker Tresp. Large-scale factorization of type-constrained multi-relational data. In *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*, pages 18–24. IEEE, 2014.

- [12] Denis Krompaß, Stephan Baier, and Volker Tresp. Type-constrained representation learning in knowledge graphs. In *International Semantic Web Conference*, pages 640–655. Springer, 2015.
- [13] Ni Lao, Tom Mitchell, and William W Cohen. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 529–539. Association for Computational Linguistics, 2011.
- [14] Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL), Berlin, Germany, August. Association for Computational Linguistics*, 2016.
- [15] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [16] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.
- [17] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 809–816, 2011.
- [18] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104:11–33, 2016.
- [19] Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. Holographic embeddings of knowledge graphs. *AAAI*, 2016.
- [20] Madhav Nimishakavi, Uday Singh Saini, and Partha Talukdar. Relation schema induction using tensor factorization with side information. *EMNLP*, 2016.
- [21] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. Relation extraction with matrix factorization and universal schemas. 2013.
- [22] Tim Rocktaschel, Sameer Singh, and Sebastian Riedel. Injecting logical background knowledge into embeddings for relation extraction. In *NAACL*, 2015.
- [23] Burr Settles. *Active Learning*. Morgan & Claypool, 2012. ISBN 978-1608457250.
- [24] Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. Representation learning of knowledge graphs with entity descriptions. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [25] Ruobing Xie, Zhiyuan Liu, and Maosong Sun. Representation learning of knowledge graphs with hierarchical types. In *IJCAI*, 2016.

## A Proofs

*Proof of Theorem 1.* In order to prove the result, it suffices to show that  $\mathcal{F}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$  in Equation (7) is submodular. Then Algorithm 2 can find a  $(1 - 1/e)$ -approximation of the global optimum of  $\mathcal{F}$ .

To demonstrate that the objective  $\mathcal{F}$  is submodular, we need to show that for  $L'' \subseteq L' \subseteq L$  and for  $l = (r_l, e_l) \in L \setminus L'$ ,

$$\mathcal{F}(L'' \cup l) - \mathcal{F}(L'') \geq \mathcal{F}(L' \cup l) - \mathcal{F}(L')$$

Since addition preserves submodularity and the weights  $w_{\mathcal{C}}, w_{\mathcal{D}}, w_{\mathcal{R}}$  are non-negative, it suffices to show that each term in  $\mathcal{F}$  is submodular.

1. In order to prove that,  $\mathcal{C}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$  is submodular, we verify that

$$\begin{aligned} \frac{(|R_{L'' \cup l}| - |R_{L''}|)}{|R|} &\geq \frac{(|R_{L' \cup l}| - |R_{L'}|)}{|R|}, \\ \frac{(|E_{L'' \cup l}| - |E_{L''}|)}{|E|} &\geq \frac{(|E_{L' \cup l}| - |E_{L'}|)}{|E|} \end{aligned}$$

$l = (r_l, e_l) \in L \setminus L'$ . Therefore, at least one of  $r_l$  or  $e_l$  is not included in  $L'$  and hence it is not included in  $L''$ . Note that for each of the above lines the difference can be either +1 or 0 and since  $L'' \subset L'$  the LHS is always greater than or equal to the RHS and the inequality holds.

2. Next we consider the diversity term. The above argument directly holds and hence it is submodular.
3. In order to show that  $-\mathcal{R}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$  is submodular, note that when taking the difference between  $\mathcal{R}(L'' \cup l)$  and  $\mathcal{R}(L'')$  the terms that correspond to both entities (relations) being in  $L''$  cancel out. The same holds for  $\mathcal{R}(L' \cup l) - \mathcal{R}(L')$ . Therefore, we have that

$$\begin{aligned} \mathcal{R}(L'' \cup l) - \mathcal{R}(L'') &= \sum_{r_1 \in l, r_2 \in L''} \|h_{r_1} - h_{r_2}\| + \sum_{e_1 \in l, e_2 \in L''} \|h_{e_1} - h_{e_2}\|, \\ \mathcal{R}(L' \cup l) - \mathcal{R}(L') &= \sum_{r_1 \in l, r_2 \in L'} \|h_{r_1} - h_{r_2}\| + \sum_{e_1 \in l, e_2 \in L'} \|h_{e_1} - h_{e_2}\|. \end{aligned}$$

Since  $L'' \subseteq L'$ ,

$$\mathcal{R}(L'' \cup l) - \mathcal{R}(L'') \leq \mathcal{R}(L' \cup l) - \mathcal{R}(L'), \text{ hence}$$

$$- [\mathcal{R}(L'' \cup l) - \mathcal{R}(L'')] \geq - [\mathcal{R}(L' \cup l) - \mathcal{R}(L')].$$

This concludes the proof. ■